Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the Application:

Listing of Claims:

Claim 1 (Currently amended): A method for caching and accessing access rights to at least one resource in a distributed computing system, the method comprising:

accessing, by a software agent, a directory service, wherein the agent is located on a deputization point coupled to the directory service, and wherein the directory service comprises the access rights of a software principal to a resource;

updating, by the agent, the access rights ~~to~~ in an access control list cache, wherein the access control list cache is coupled to the deputization point and to the principal;

receiving, at the access control list cache, a request from the principal for the access rights stored in the access control list cache;

retrieving, ~~by~~ from the access control list cache, the access rights;

forwarding, to the principal, the access rights; ~~and~~

delegating one or more of the principal's access rights to at least one software entity; and

accessing the resource, by the software entity, using the delegated access rights without requiring intervention of the principal to authenticate access requests by the software entity, wherein tasks can be accomplished by the software entity without control by the principal.

~~deputizing the principal to enable the principal to delegate the rights to at least one software entity, wherein the at least one software entity can exercise the rights due to the delegation.~~

Claim 2 (Currently amended): The method of claim 1, wherein the access control list cache is comprised of a first table comprising the principal that has access to the resource.

Claim 3 (Currently amended): The method of claim 1, wherein the access control list cache is comprised of a second table comprising the access rights of the principal to the resource.

Claim 4 (Original): The method of claim 1, wherein the access control list cache is comprised of a third table comprising a cached access to the resource object.

Claim 5 (Currently amended): The method of claim 2 further comprising invoking, by the directory service, a resource manager, if the first table does not contain the principal that has access to the resource, wherein the resource manager is coupled to the directory service and comprises access information and access rights of the principal to the resource.

Claim 6 (Currently amended): The method of claim 5 further comprising mapping, by the resource manager, an access control of the access rights in the resource manager to an access control of the rights in the directory service.

Claim 7 (Currently amended): The method of claim 6 further comprising updating, by the resource manager, the mapped access control of the access rights to the access control list cache.

Claims 8 and 9 (Cancelled).

Claim 10 (Currently amended): The method of claim 1, further comprising at least one of the following actions from the group consisting of:

asynchronously updating, by the agent to the access control list cache, the access rights, when the access rights are added to the directory service;

asynchronously updating, by the agent to the access control list cache, the access rights, when the access rights are removed from the directory service;

asynchronously updating, by the agent to the access control list cache, the access rights, when the request from the principal is received;

synchronously updating, by the agent to the access control list cache, the access rights, when the access rights are added to the directory service;

synchronously updating, by the agent to the access control list cache, the access rights, when the access rights are removed from the directory service;

synchronously updating, by the agent to the access control list cache, the access rights, when the request from the principal is received;

updating, at a scheduled time, the access rights by the agent to the access control list cache; and

updating, after a time to live has expired, the access rights by the agent to the access control list cache.

Claim 11 (Currently amended): A distributed computing system supporting access control caching, the system comprises:

a plurality of computers, each having a memory and a processor;

a plurality of communication links connecting the plurality of computers;

a principal located on a first one of the computers;

an agent located on a second one of the computers;

a resource located on a third one of the computers;

a first set of access rights located on a fourth one of the computers;

a second set of access rights located on a fifth one of the computers;

means for accessing, by the agent, the first set of access rights of the principal to the resource;

means for updating, by the agent, the first set of access rights to an access control list cache, wherein the access control list cache is located on a sixth one of the computers;

means for receiving, at the access control list cache, a request from the principal for the first set of access rights;

means for retrieving, by the access control list cache, the first set of access rights;

means for forwarding, to the principal, the first set of access rights; and

means for providing, to the principal, a deputization certificate adapted for enabling the principle to copy ~~its~~ one or more of the principal's access rights to at least one software entity.

Claim 12 (Currently amended): The system of claim 11 further comprises means for invoking the second set of access rights, if the first set of access rights is not located on the fourth one of the computers.

Claim 13 (Currently amended): The system of claim 12 further comprises means for mapping an access control of the ~~of the~~ second set of access rights to an access control of the first set of access rights.

Claim 14 (Currently amended): The system of claim 13 further comprises, means for updating the access control list cache with the mapped access control of the first set of access rights.

Claim 15 (Currently amended): A computer storage medium having a configuration that represents data and instructions which will cause performance of method steps for caching and accessing access rights in a distributed computing system, the method comprising:

accessing, by a software agent, a directory service, wherein the agent is located on a deputization point coupled to the directory service having the access rights of at least one principal to at least one resource;

updating, by the agent, the access rights to an access control list cache, wherein the access control list cache is coupled to the deputization point, and wherein the access control list cache is coupled to the principal;

receiving, at the access control list cache, a request from the principal for the access rights;

retrieving, by the access control list cache, the access rights; ~~and~~

forwarding, to the principal, the access rights;

forwarding, to the principal, a deputization credential empowering the principal to deputize software entities; and

deputizing, by the principal, at least one of the software entities, wherein the software entity can exercise one or more of the principal's access rights due to the deputization.

Claim 16 (Currently amended): The configured storage medium of claim 15 further comprising invoking, by the directory service, a resource manager, if the access control list cache does not contain one of the access rights, wherein the resource manager is coupled to the directory service, and wherein the resource manager comprises the one right.

Claim 17 (Currently amended): The configured storage medium of claim 16 further comprising mapping, by the resource manager, an access control of the one right to an access control of the access rights.

Claim 18 (Currently amended): The configured storage medium of claim 17 further comprising updating, by the resource manager, the mapped access control of the access rights to the access control list cache.

Claim 19-22 (Cancelled).

Claim 23 (Currently Amended): A method for controlling access within a computer system using deputization, the method comprising:

receiving an access authorization request at a deputization point from a principal, wherein the access authorization request requests validation of the principal's identity;

determining whether to validate the principal based on the access authorization request;

identifying ~~an access authorization level~~ one or more resource access permissions for the principal if the principal is validated, wherein the resource access permissions enable the principal to access one or more resources; and

providing the principal with deputizing authority at the identified access authorization level, wherein the deputizing authority comprises a deputization credential that enables the principal to give at least one software entity within the computer system a level of ~~access authorization~~ resource access permission equal to or lesser than the principal's ~~access authorization level~~ resource access permissions.

Claim 24 (Previously presented): The method of claim 23 wherein determining whether to validate the principal includes comparing information present in the access authorization request to a plurality of access rights contained in an access control list cache.

Claim 25 (Previously presented): The method of claim 24 further comprising:

invoking a resource manager if the access control list cache does not contain an access right associated with the access authorization request;

locating the access right associated with the access authorization request; and

mapping the access right into the plurality of access rights.

Claim 26 (Currently amended): The method of claim 23 further comprising deputizing, by the principal, a first software entity, wherein the first software entity has a level of ~~access authorization~~ resource access permission equal to or lesser than the principal's ~~access authorization level~~ resource access permissions.

Claim 27 (Currently amended): The method of claim 26 wherein deputizing includes defining ~~a level of access authorization and~~ a lifespan of the deputization.

Claim 28 (Currently amended): The method of claim 26 further comprising deputizing, by the first software entity, a second software entity, wherein the second software entity has a level of ~~access authorization~~ resource access permission equal to or lesser than the first software entity's ~~access authorization~~ level of resource access permission.

29. (New) A computer-executable method for delegating permission from a software principal to a software deputy within a computer network to access at least one resource that is accessible to the principal, the method comprising:

receiving a request from the principal for a deputy credential, wherein the request includes the principal's identity and at least one permission to be assigned to the deputy;

sending the deputy credential to the principal, wherein the deputy credential enables the principal to assign the permission to the resource to the deputy;

receiving a deputization request from the principal to assign the permission to the deputy; and

assigning the permission to the deputy, wherein the deputy can independently access the resource using the assigned permission without being controlled by the principal.

30. (New) The method of claim 29 further comprising imposing a lifespan on the assignment of the permission, wherein the assignment will expire at the end of the lifespan.

31. (New) The method of claim 29 further comprising imposing a lifespan on the deputy, wherein the deputy will terminate at the end of the lifespan.

32. (New) The method of claim 29 further comprising:

determining if a deputy identified in the deputization exists; and

creating the deputy if the deputy does not exist.

33. (New) The method of claim 32 further comprising identifying a start time in the deputization request for assigning the permission to the deputy, wherein the permission is not assigned to the deputy until the start time.

34. (New) The method of claim 33 wherein the principal is terminated in the computer network prior to the start time.

35. (New) The method of claim 29 further comprising verifying that the principal is permitted to access the resource prior to sending the deputy credential to the principal.

36. (New) The method of claim 29 wherein the deputy is in a namespace that is not accessible to the principal, and wherein the deputy can use the permission to access a resource in the namespace that is not accessible to the principal.

37. (New) The method of claim 29 wherein the request from the principal for a deputy credential includes a plurality of permissions to be assigned to the deputy, and wherein the deputy credential sent to the principal permits the principal to assign only a portion of the plurality of permissions to the deputy.

38. (New) The method of claim 29 further comprising

receiving a second request from the principal for a second deputy credential, wherein the request includes the principal's identity and at least a second permission to be assigned to the deputy;

sending the second deputy credential to the principal; and

assigning the second permission contained in the second deputy credential to the deputy, wherein the deputy includes permissions from both the deputy credential and the second deputy credential.